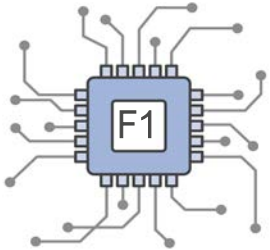
FPGA Accelerated Computing Using AWS F1 Instances

Applications and development environment



David Pellerin, Amazon Web Services

HotChips 2017

August 22, 2017

amazon web services

Why Accelerated Computing in the Cloud?

Parallelism increases throughout…

CPU: High speed, low efficiency

GPU/FPGA: High throughput, high efficiency

GPUs and FPGAs can provide massive parallelism and higher efficiency than CPUs for certain categories of applications

Compelling Use-Cases for Acceleration

Deep Learning Training and Inference

Video and Image Processing

Engineering Simulations

Financial Computing

Molecular Dynamics

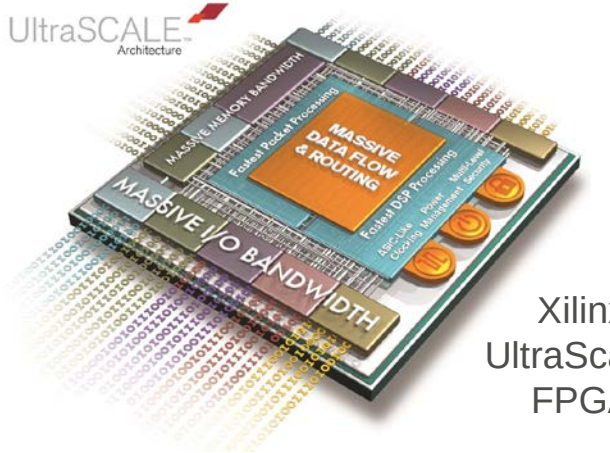
VR Content Rendering

Accelerated Search and Databases

Many More

GPU and FPGA for Accelerated Computing

NVIDIA GPU



Xilinx
UltraScale+
FPGA

P2: GPU-accelerated computing

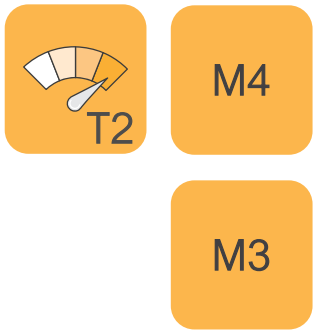
- Enabling a high degree of parallelism – each GPU has thousands of cores
- Consistent, well documented set of APIs (CUDA, OpenACC, OpenCL)
- Supported by a wide variety of ISVs and open source frameworks

F1: FPGA-accelerated computing

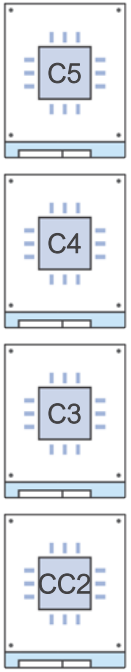
- Massively parallel – each FPGA includes millions of parallel system logic cells
- Flexible – no fixed instruction set, can implement wide or narrow datapaths
- Programmable using available, cloud-based FPGA development tools

AWS Compute Instance Types

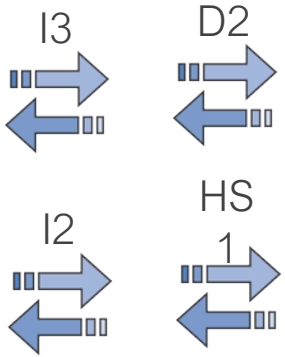
General purpose



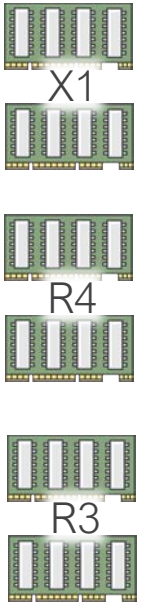
Compute optimized



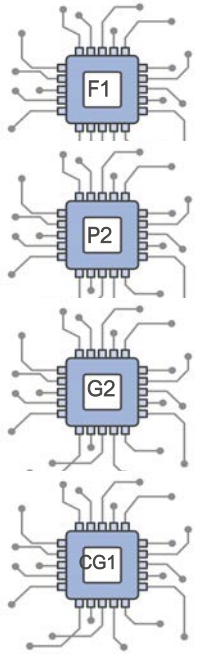
Storage and IO optimized



Memory optimized



GPU and FPGA accelerated



2017

2016

2013

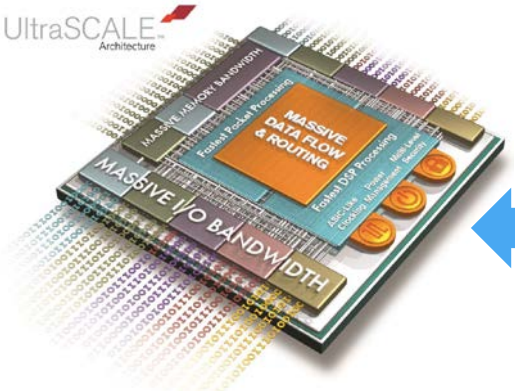
2011

FPGA Acceleration in the AWS Cloud: Goals

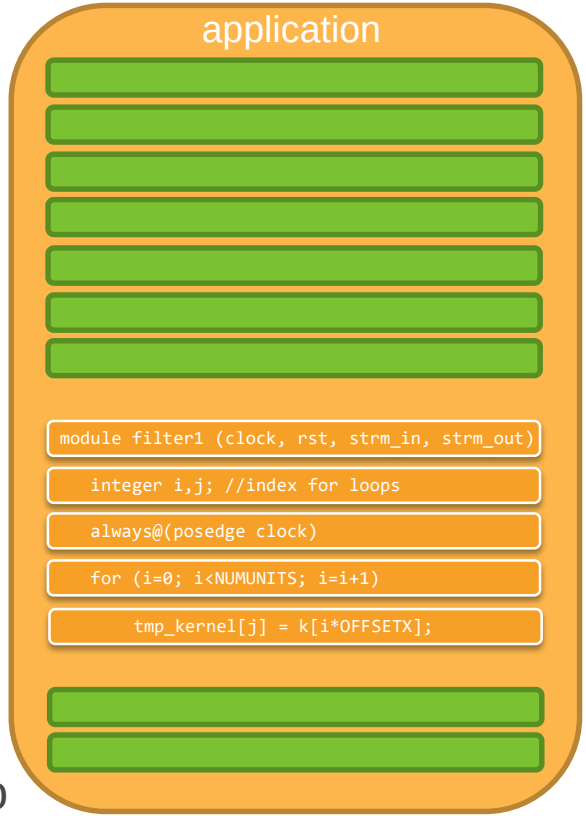
- **Make FPGAs available as standard AWS instances** to a large community of developers, and to millions of potential end-customers
- **Simplify the development process** by providing cloud-based FPGA development tools
- **Allow developers to focus on algorithm design**, by abstracting FPGA I/O using well-defined interfaces
- **Provide a Marketplace for FPGA applications**, providing more choice and easy access for all AWS customers

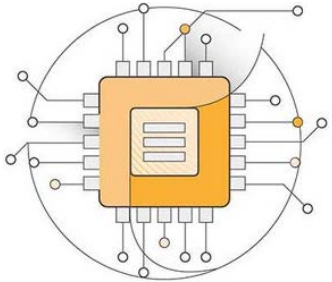
How FPGA Acceleration Works on AWS

FPGA handles compute-intensive, deeply pipelined, hardware-accelerated operations



Dedicated PCIe and ring connections also allow communication between up to 8 FPGAs, at up to 400Gbps

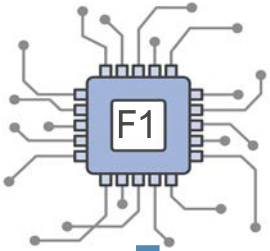




CPU handles the rest

Data is transferred to and from the FPGA via PCIe

Guiding principle: allow FPGAs to be included in a customer's deployment as easily as any other AWS instance type or service

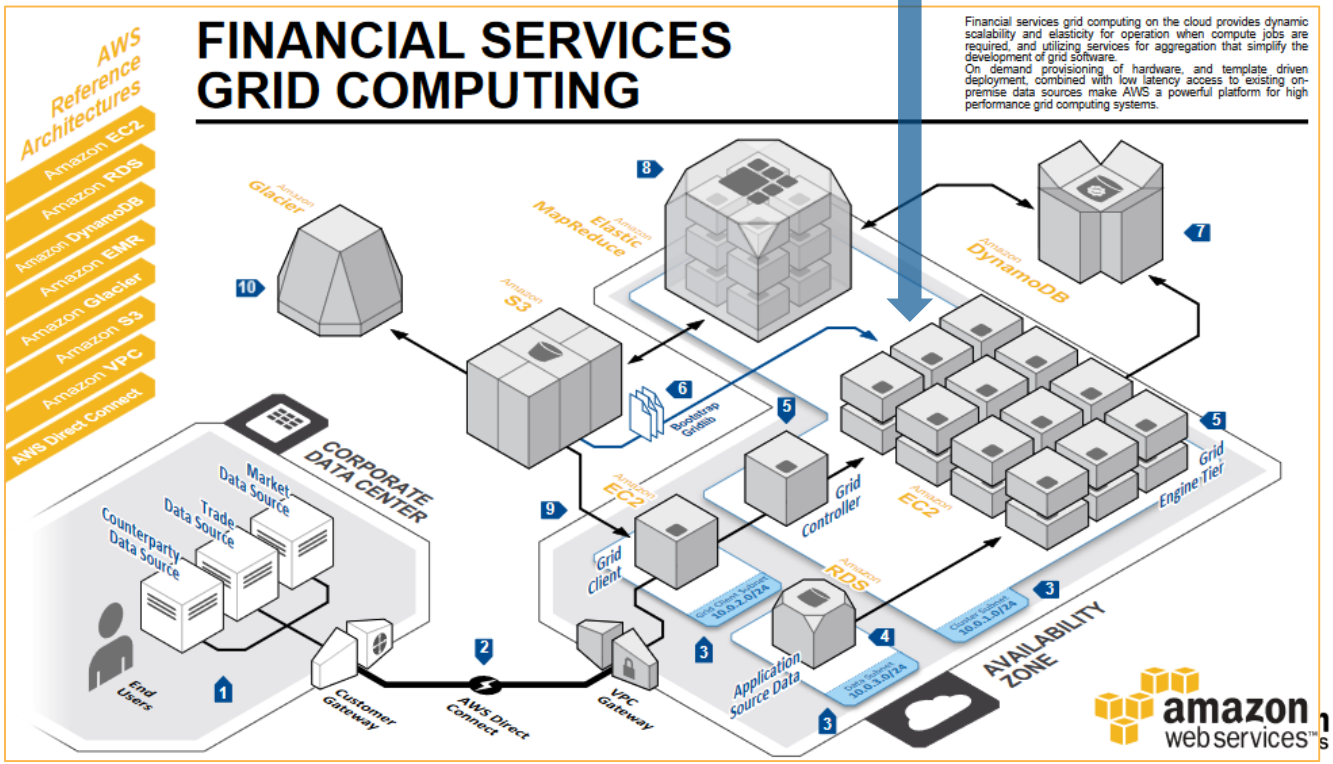


Important concepts:

Region
Availability Zone (AZ)
Virtual Private Cloud (VPC)
Elastic Compute Cloud (EC2)
Amazon Machine Image (AMI)
EC2 Instance
AWS Marketplace

Additional for F1:

FPGA Developer AMI
Amazon FPGA Image (AFI)



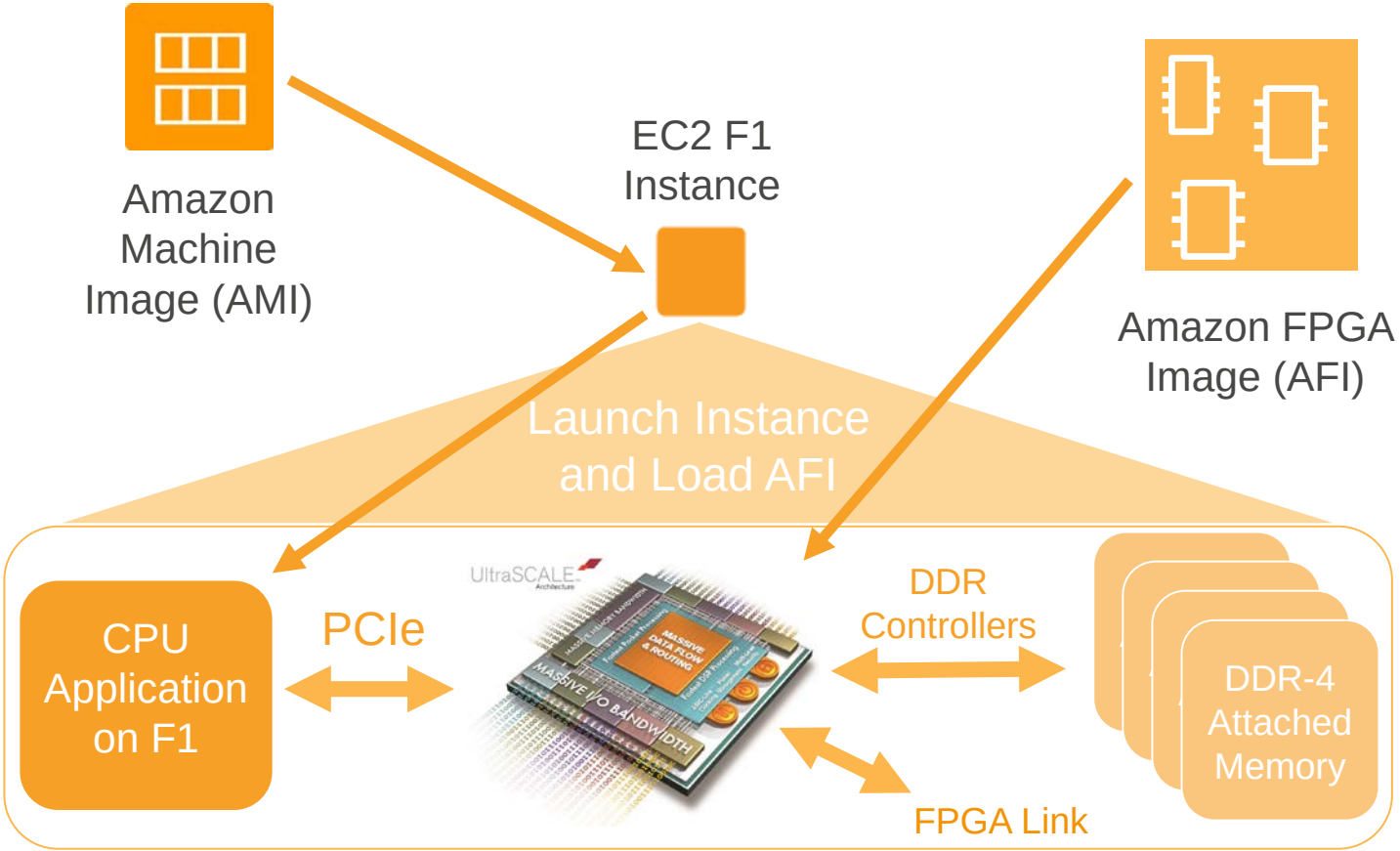
F1 Instances

Model	FPGAs	vCPU	Mem (GiB)	SSD Storage (GB)	Networking Performance
f1.2xlarge	1	8	122	470	Up to 10 Gigabit
f1.16xlarge	8	64	976	4 x 940	20 Gigabit

For F1.16xlarge instances, the dedicated PCI-e fabric lets the FPGAs share the same memory space and communicate with each other across the fabric at up to 12 GBps in each direction. The FPGAs within the F1.16xlarge share access to a 400 Gbps bidirectional ring for low-latency, high bandwidth communication.

- Up to eight Xilinx UltraScale Plus VU9P FPGAs per F1 instance
- Each FPGA includes
  - Local 64 GiB DDR4 ECC protected memory
  - Dedicated PCIe x16 connections, and an up to 400Gbps bidirectional ring connection for high-speed streaming
  - Approximately 2.5 million logic elements, and approximately 6,800 Digital Signal Processing (DSP) engines

FPGA Acceleration Using F1



An F1 instance can have any number of AFIs

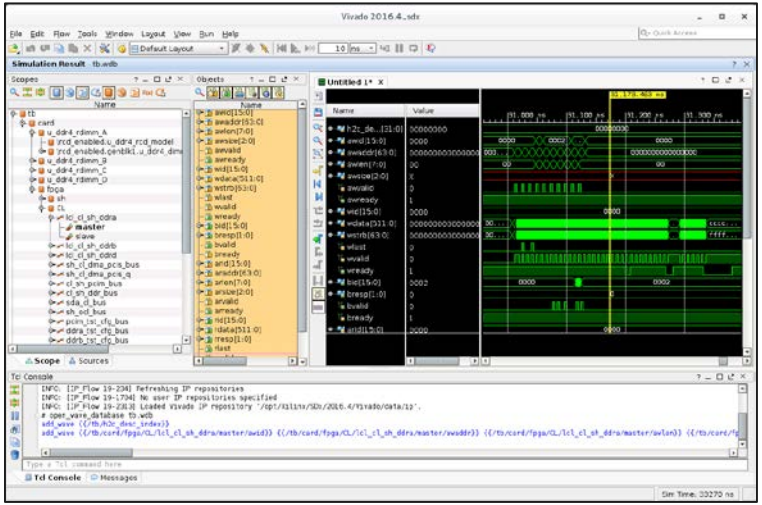
An AFI can be loaded into the FPGA in less than 1 second

Developing Applications for F1

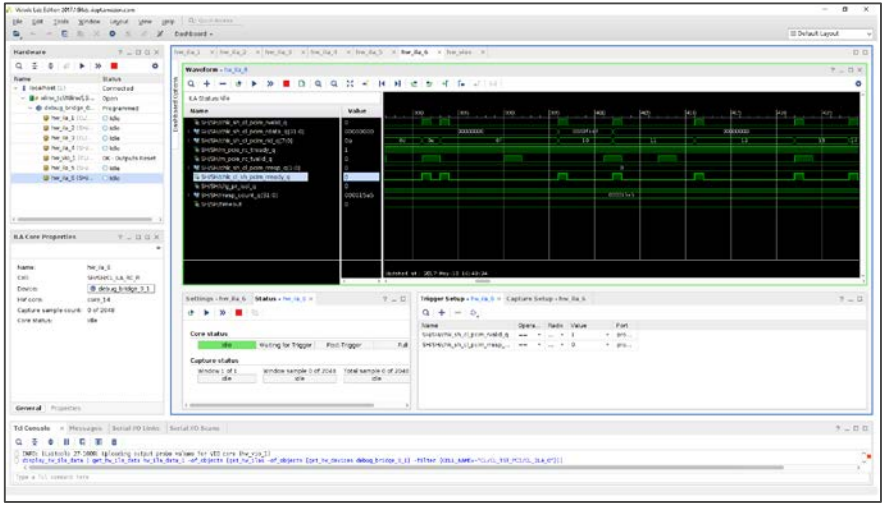
The F1 Development AMI

Use Xilinx Vivado and a hardware description language (Verilog or VHDL for RTL, or optionally using the OpenCL framework) with the HDK to describe and simulate your custom FPGA logic

Xilinx Vivado for custom logic development



Virtual JTAG for interactive debugging



FPGA Developer AMI

Sold by: Amazon Web Services

The FPGA (field programmable gate array) AMI is a supported and maintained CentOS Linux image provided by Amazon Web Services. The AMI is pre-built with FPGA development tools and run time tools required to develop and use custom FPGAs for hardware acceleration. The FPGA developer AMI includes a prepackaged tool development environment, with scripts and tools for simulating your FPGA design, compiling code, building and registering your AFI (Amazon FPGA Image). Developers can deploy the FPGA developer AMI on an Amazon EC2 instance and quickly provision the resources they need to write... Read more

Customer Rating	★★★★★ (0 Customer Reviews)
Latest Version	1.2.1
Operating System	Linux/Unix, CentOS 7.3
Delivery Method	64-bit Amazon Machine Image (AMI) (Read more)
Support	See details below
AWS Services Required	Amazon EC2, Amazon EBS
Highlights	▪ Xilinx Vivado 2017.1 and 2016.4 SDx - Free license for F1 FPGA development ▪ AWS Integration - includes packages and configurations that provide tight integration with Amazon Web Services

Product Description

The FPGA (field programmable gate array) AMI is a supported and maintained CentOS Linux image provided by Amazon Web Services. The AMI is pre-built with FPGA development tools and run time tools required to develop and use custom FPGAs for hardware acceleration. The FPGA developer AMI includes a prepackaged tool development environment, with scripts and tools for simulating your FPGA design, compiling code, building and registering your AFI (Amazon FPGA Image). Developers can deploy the FPGA developer AMI on an Amazon EC2 instance and quickly provision the resources they need to write and debug FPGA designs in the cloud. The AMI is designed to provide a stable, secure, and high performance development environment. The FPGA AMI is provided at no additional charge to Amazon EC2 users.

Continue You will have an opportunity to review your order before launching or being charged.

Pricing Information

Use the Region dropdown selector to see software and infrastructure pricing information for the chosen AWS region.

For Region

US East (N. Virginia)

Pricing Details

Software pricing is based on your chosen options, such as subscription term and AWS region. Infrastructure prices are estimates only. Final prices will be calculated according to actual usage and reflected on your monthly report.

1 Software Pricing

The data below shows pricing per instance for services hosted in US East (N. Virginia).

FPGA Developer AMI - Hourly

EC2 Instance Type	Software /hr	EC2 /hr	Total /hr
c4.4xlarge	$0.00	$0.796	**$0.796**
c4.8xlarge	$0.00	$1.591	**$1.591**
m4.2xlarge	$0.00	$0.431	**$0.431**
m4.4xlarge	$0.00	$0.862	**$0.862**
m4.10xlarge	$0.00	$2.155	**$2.155**
m4.16xlarge	$0.00	$3.447	**$3.447**
t2.2xlarge	$0.00	$0.376	**$0.376**
f1.2xlarge	$0.00	$1.65	**$1.65**
f1.16xlarge	$0.00	$13.20	**$13.20**
r4.xlarge	$0.00	$0.266	**$0.266**
r4.2xlarge	$0.00	$0.532	**$0.532**
r4.4xlarge	$0.00	$1.064	**$1.064**
r4.8xlarge	$0.00	$2.128	**$2.128**
r4.16xlarge	$0.00	$4.256	**$4.256**

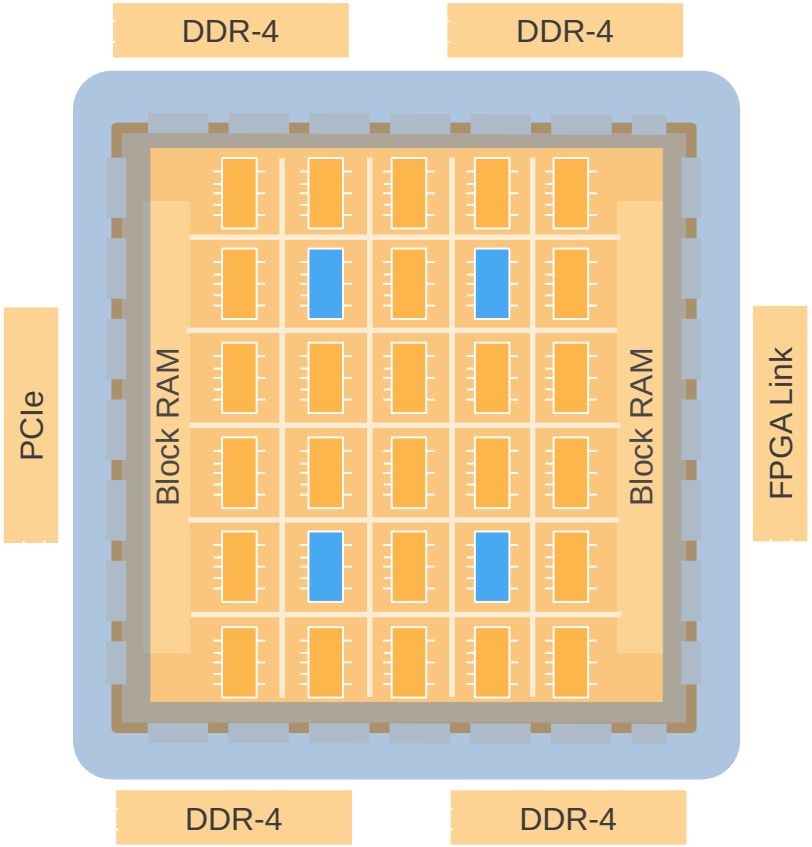
1 Software Pricing

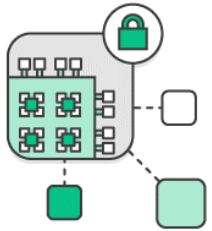
The data below shows pricing per instance for services hosted in US East (N. Virginia).

FPGA Developer AMI - Hourly

EC2 Instance Type	Software /hr	EC2 /hr	Total /hr
c4.4xlarge	$0.00	$0.796	**$0.796**
c4.8xlarge	$0.00	$1.591	**$1.591**
m4.2xlarge	$0.00	$0.431	**$0.431**
m4.4xlarge	$0.00	$0.862	**$0.862**
m4.10xlarge	$0.00	$2.155	**$2.155**
m4.16xlarge	$0.00	$3.447	**$3.447**
t2.2xlarge	$0.00	$0.376	**$0.376**
f1.2xlarge	$0.00	$1.65	**$1.65**
f1.16xlarge	$0.00	$13.20	**$13.20**
r4.xlarge	$0.00	$0.266	**$0.266**
r4.2xlarge	$0.00	$0.532	**$0.532**
r4.4xlarge	$0.00	$1.064	**$1.064**
r4.8xlarge	$0.00	$2.128	**$2.128**
r4.16xlarge	$0.00	$4.256	**$4.256**

Abstracting FPGA I/O

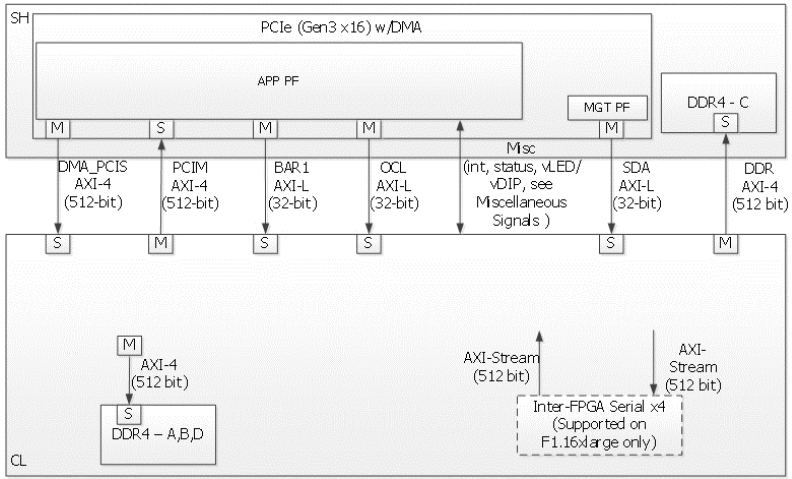




AWS FPGA Shell

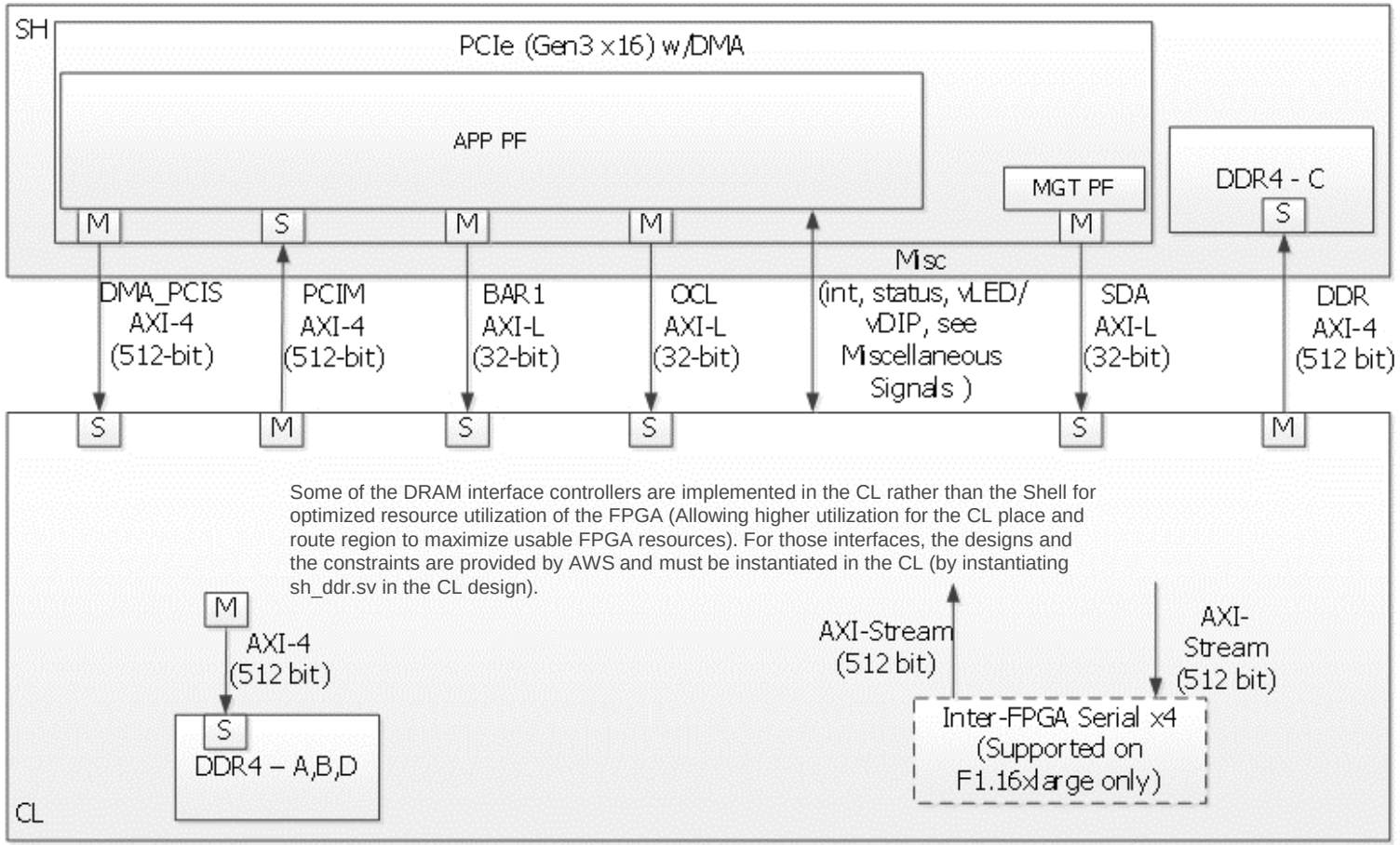
FPGA I/O is provided using standard, pre-tested, and secure I/O components, allowing FPGA developers to focus on their differentiating value

The FPGA Shell allows for faster coding of core acceleration functions by removing the need to develop I/O related FPGA hardware



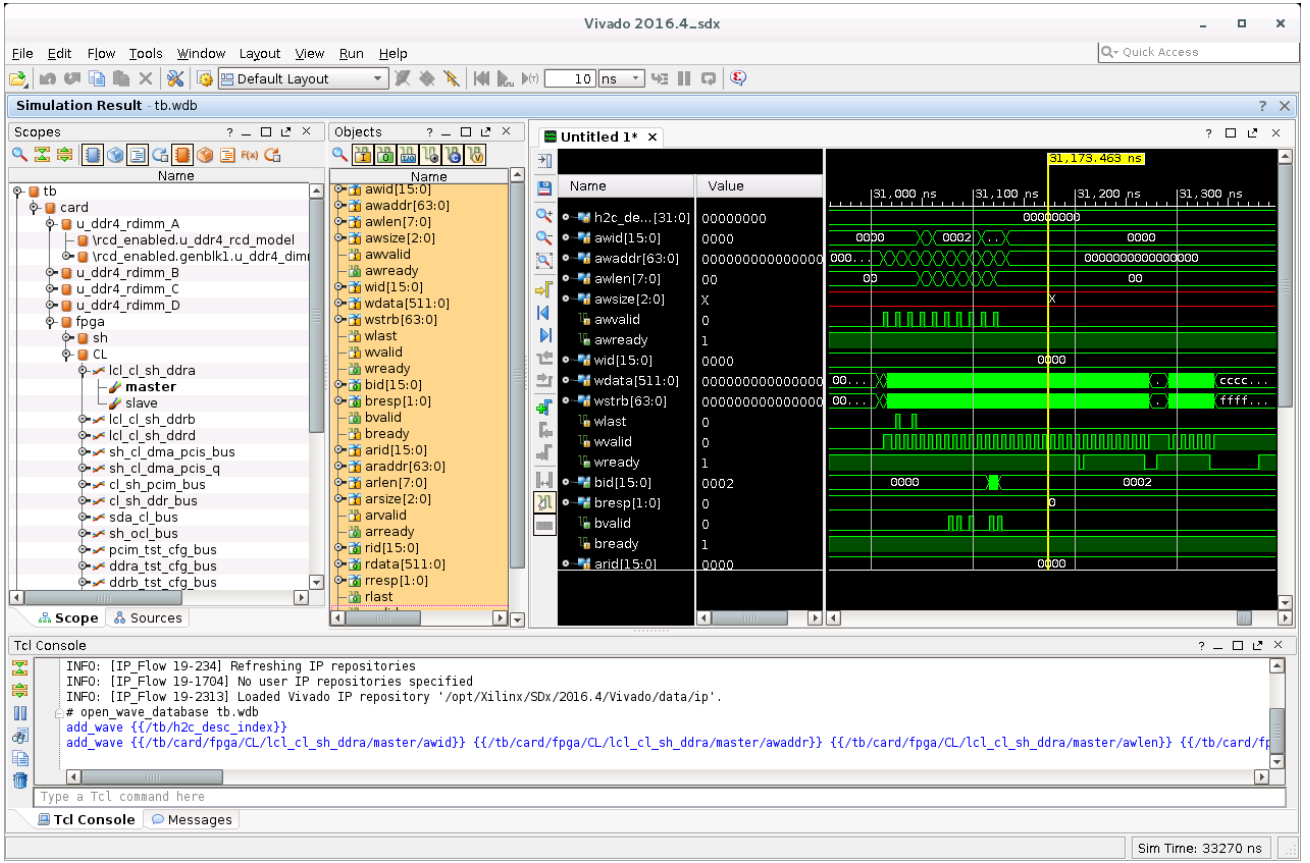
amazon web services

FPGA Shell and FPGA Custom Logic



Hardware Simulation on AWS

Run RTL simulation using the simulator of your choice, either using the AWS-provided FPGA Developer AMI, or using your choice of simulation tools



Using Build Strategies to Accelerate Development

Strategy descriptions:

```
$ ./aws_build_dcp_from_cl.sh  [-h | -H | -help] [-script <vivado_script>] [-strategy <BASIC | DEFAULT | EXPLO
```

BASIC
The basic flow in Vivado, designed to provide a good balance between runtime and Quality of Results (QOR)

EXPLORE
This is a high-effort flow which is designed to give improved QOR results at the expense of runtime

TIMING
This flow is designed for more aggressive timing optimization at the expense of runtime and congestion

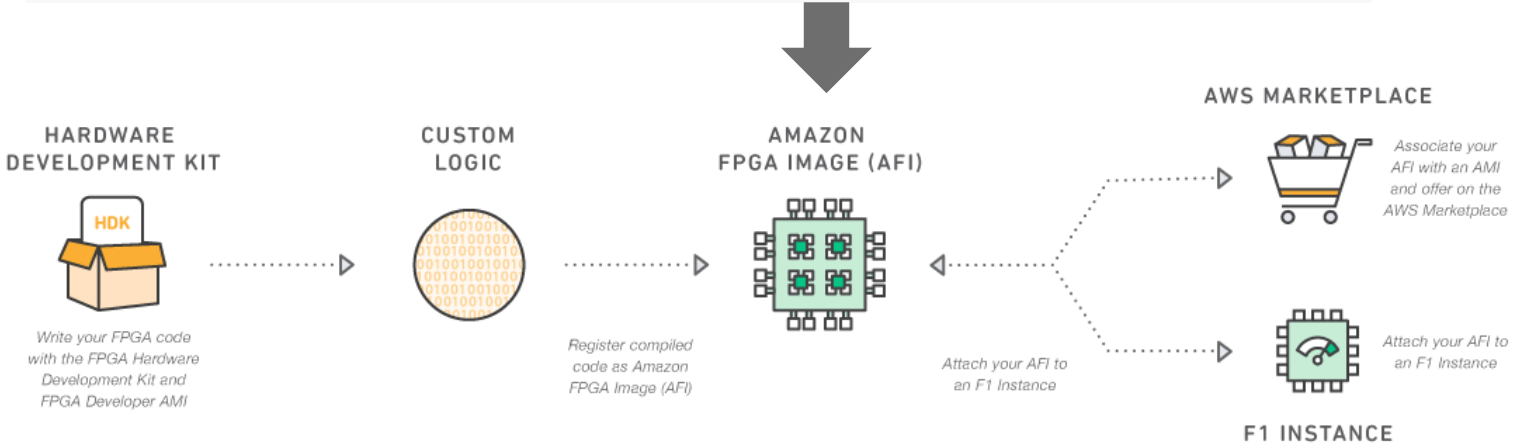
CONGESTION
This flow is designed to insert more aggressive whitespace to alleviate routing congestion

DEFAULT
This is an additional high-effort flow that results in improved QOR results for the example design at the expense of runtime

Create the Amazon FPGA Image (AFI)

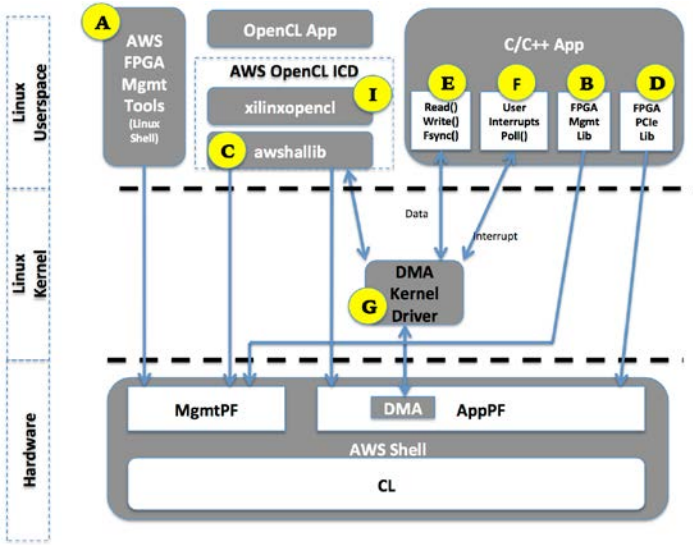
Generate an encrypted AFI using the generated DCP

```
$ aws ec2 create-fpga-image \
    --name <afi-name> \
    --description <afi-description> \
    --input-storage-location Bucket=<dcp-bucket-name>,Key=<path-to-tarball> \
    --logs-storage-location Bucket=<logs-bucket-name>,Key=<path-to-logs> \
    [ --client-token <value> ] \
    [ --dry-run | --no-dry-run ]
```



AWS FPGA SDK

- SDK includes the software runtime environment required to deploy on F1 instances and perform FPGA debugging
- Includes the drivers and tools to manage deployment of the AFIs to the F1 FPGAs, and to manage I/O from the software side
- APIs can be used to load different AFIs onto the F1 instance, without requiring an instance reboot



SDK

Management options:

[A] Shell FPGA Management Tools

[B] C-library FPGA Management

[C] OpenCL runtime library

Runtime code for I/O:

[D] FPGA PCIe Lib

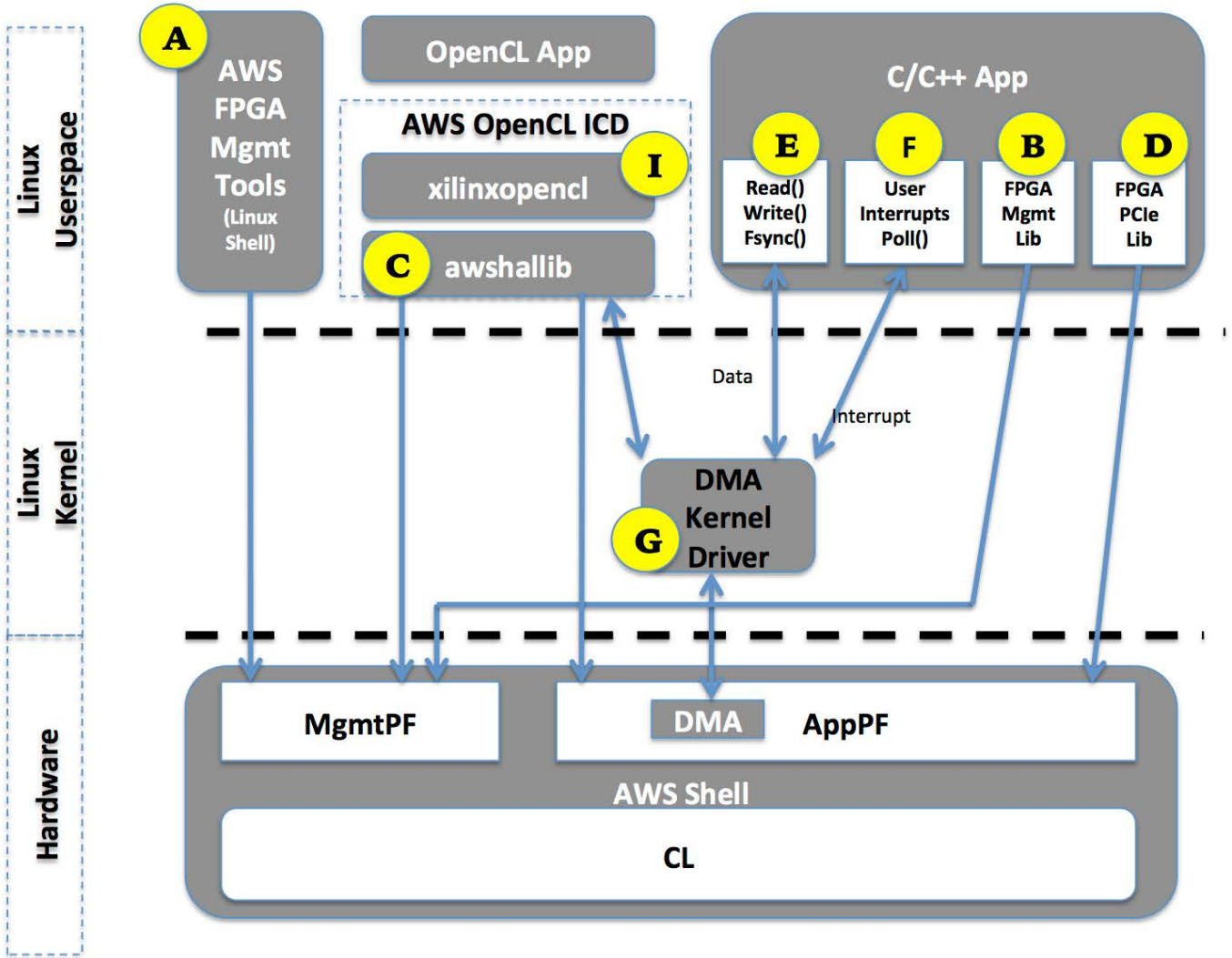
[E] DMA Interface

[F] Interrupt/Event notification

[I] OpenCL Installable Client Driver

Linux Kernel Driver:

[G] DMA Kernel Driver



AWS FPGA SDK - APIs

Management APIs

fpga-load-local-image,
fpga-clear-local-image,
fpga-describe-local,
fpga-start-virtual-jtag,
fpga-get-virtual-led,
fpga-set-virtual-dip-switch

Runtime driver library APIs

```
write_buffer = (char *)malloc(buffer_size);
read_buffer = (char *)malloc(buffer_size);
if (write_buffer == NULL || read_buffer == NULL) {
    rc = ENOMEM;
    goto out;
}

rand_string(write_buffer, buffer_size);

for (channel=0;channel < 4; channel++) {

    rc = pwrite(fd, write_buffer, buffer_size, 0x10000000 + channel*MEM_16G);

    fail_on((rc = (rc < 0)? 1:0), out, "call to pwrite failed.");

}
```

F1 Now Supports OpenCL

Application Code has two parts:

Host code

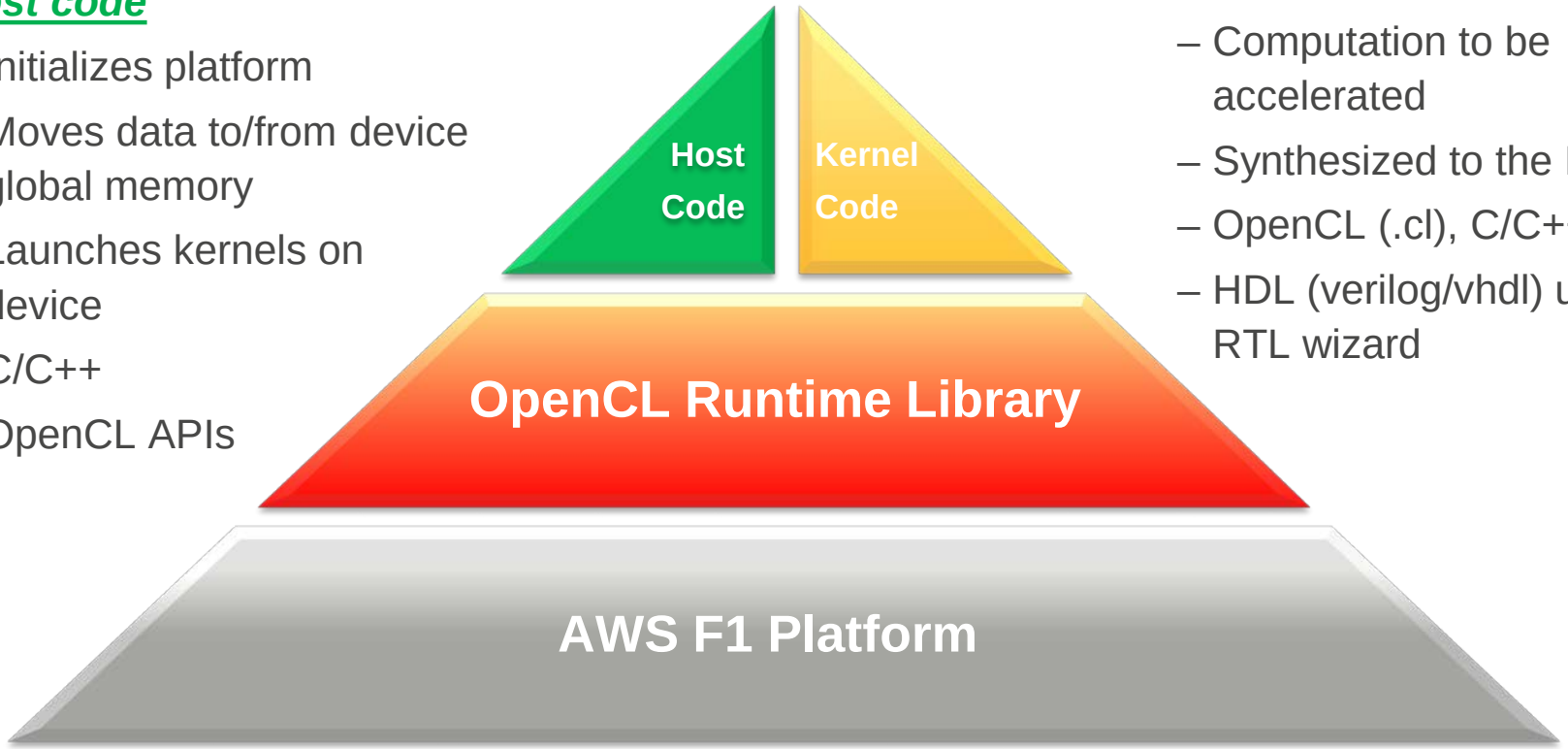
– Initializes platform

– Moves data to/from device global memory

– Launches kernels on device

– C/C++

– OpenCL APIs



Kernel code

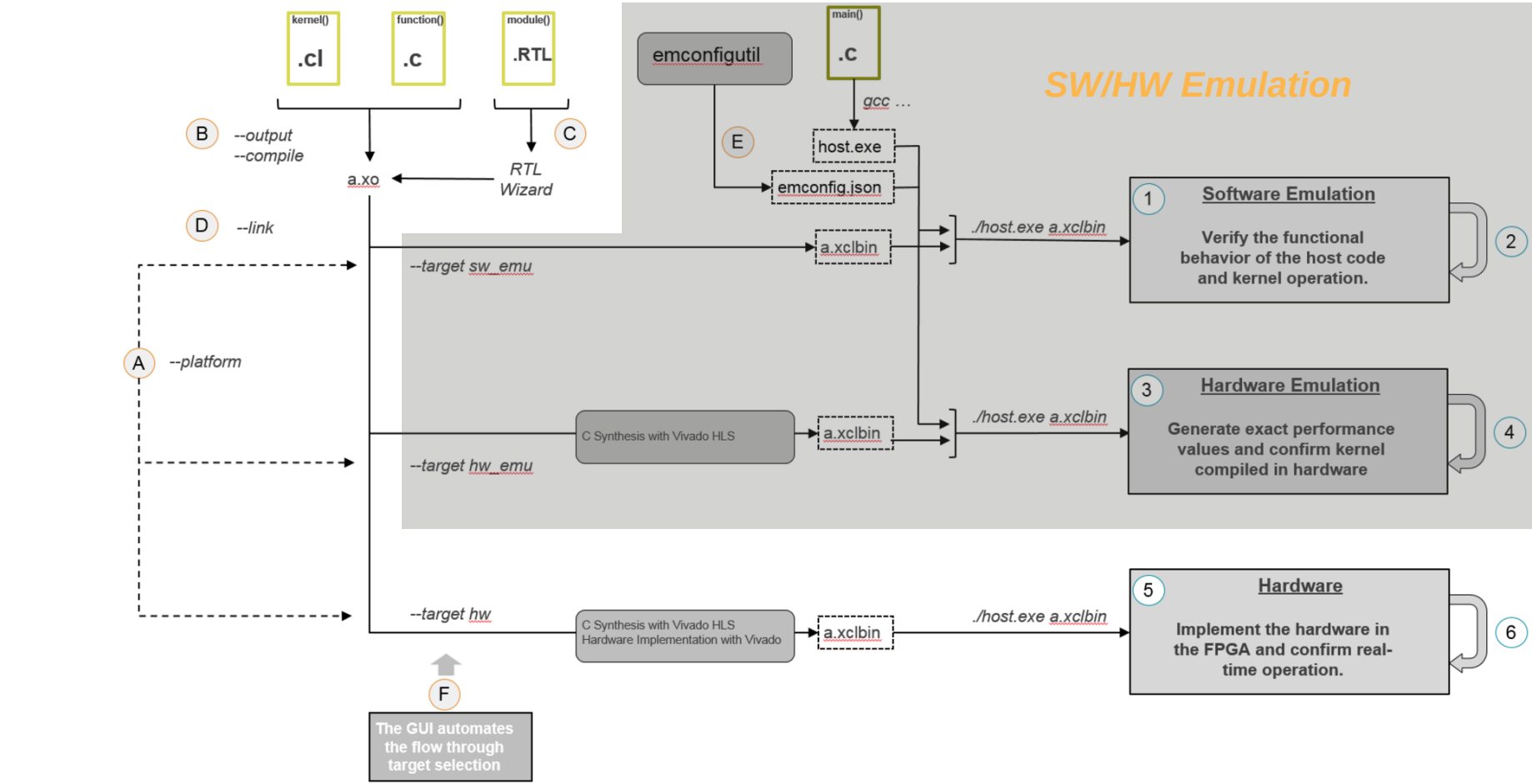
– Computation to be accelerated

– Synthesized to the FPGA

– OpenCL (.cl), C/C++

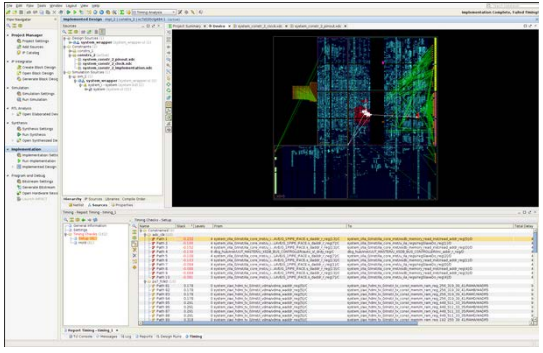
– HDL (verilog/vhdl) using RTL wizard

F1 OpenCL Design Flow

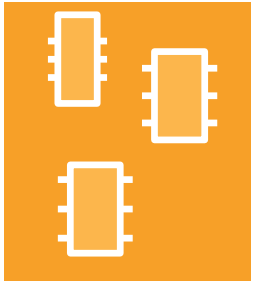


Developing Applications for F1 – AFI Creation

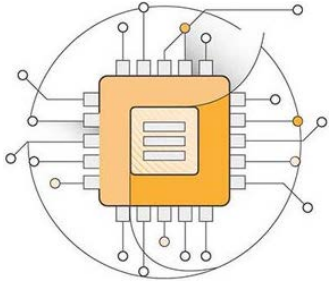
DCP



FPGA Development using Xilinx Vivado on C4 or M4 instance



Generate an Amazon FPGA Image (AFI)

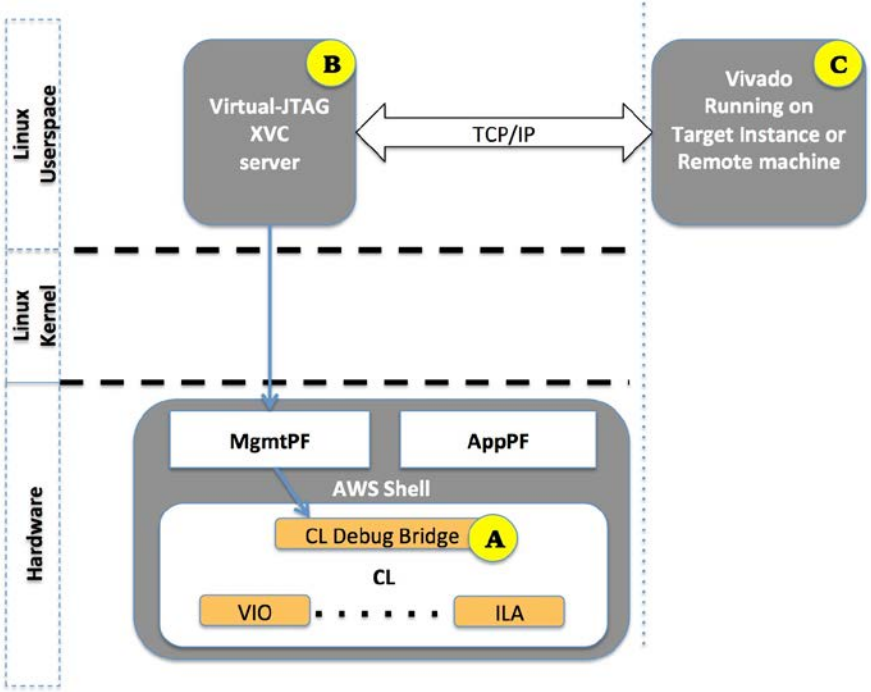


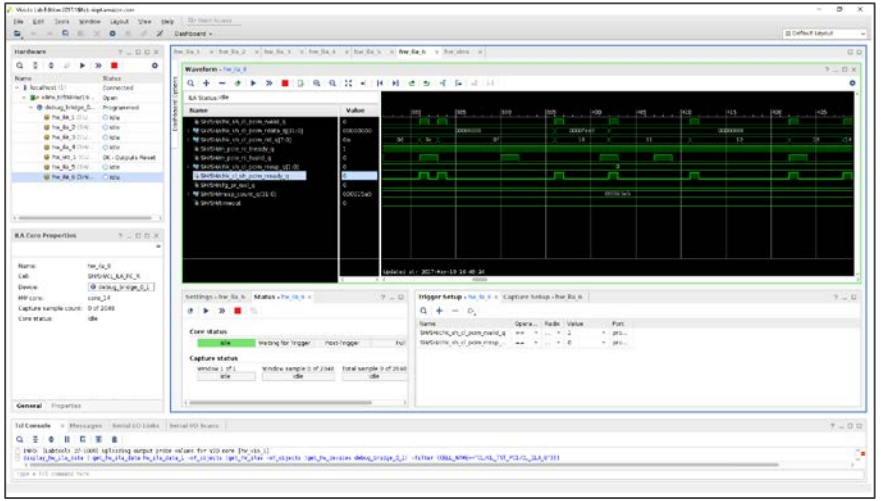
Securely deploy AFI on one or more F1 instances

```
$ aws ec2 create-fpga-image \
    --name <afi-name> \
    --description <afi-description> \
    --input-storage-location Bucket=<dcp-bucket-name>,Key=<path-to-tarball> \
    --logs-storage-location Bucket=<logs-bucket-name>,Key=<path-to-logs> \
    [ --client-token <value> ] \
    [ --dry-run | --no-dry-run ]
```

Virtual JTAG for Runtime Debugging

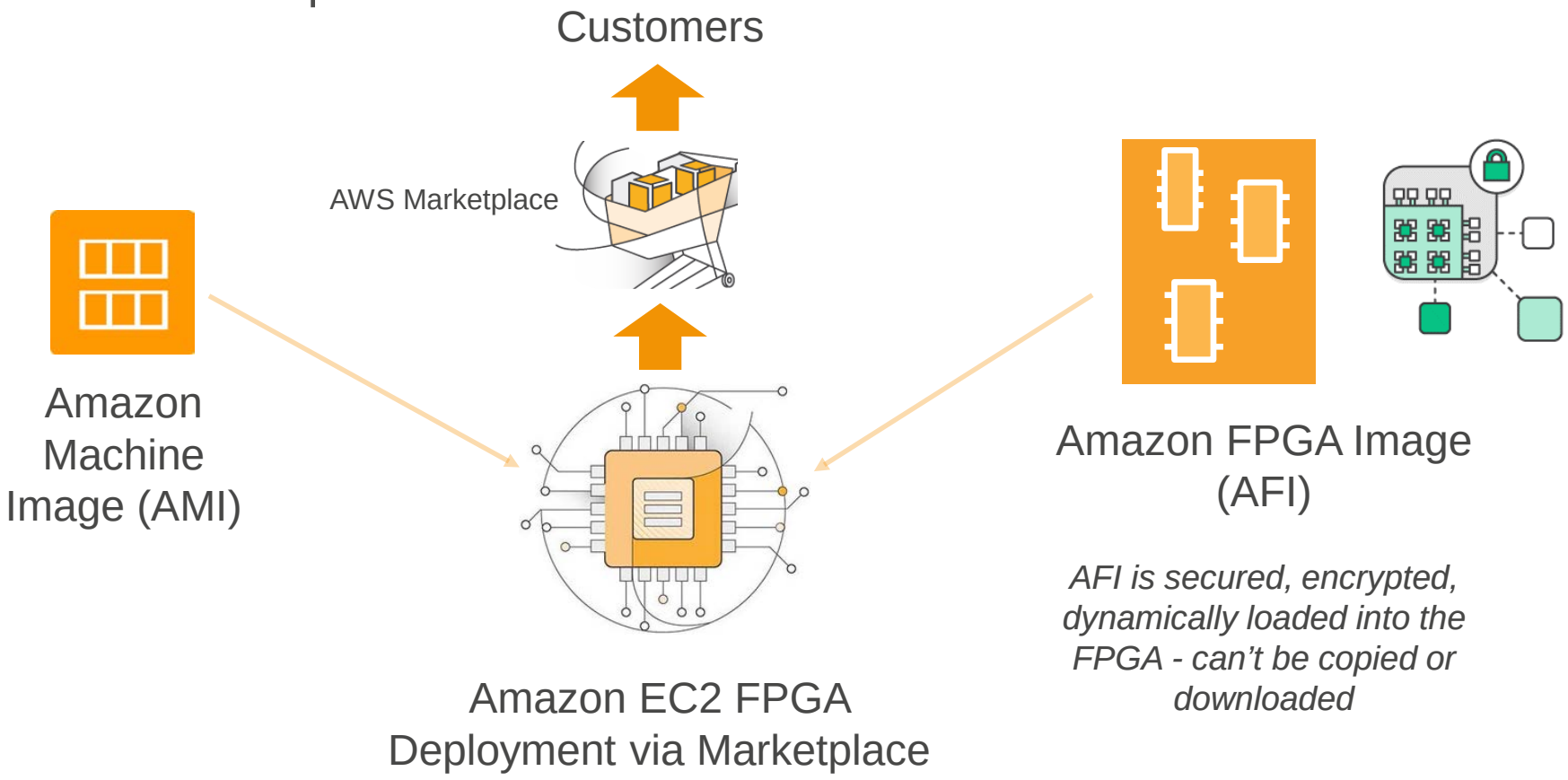
```
$ sudo fpga-start-virtual-jtag -P 10201 -S 0
Starting Virtual JTAG XVC Server for FPGA slot id 0, listening to TCP port 10201.
Press CTRL-C to stop the service.
```





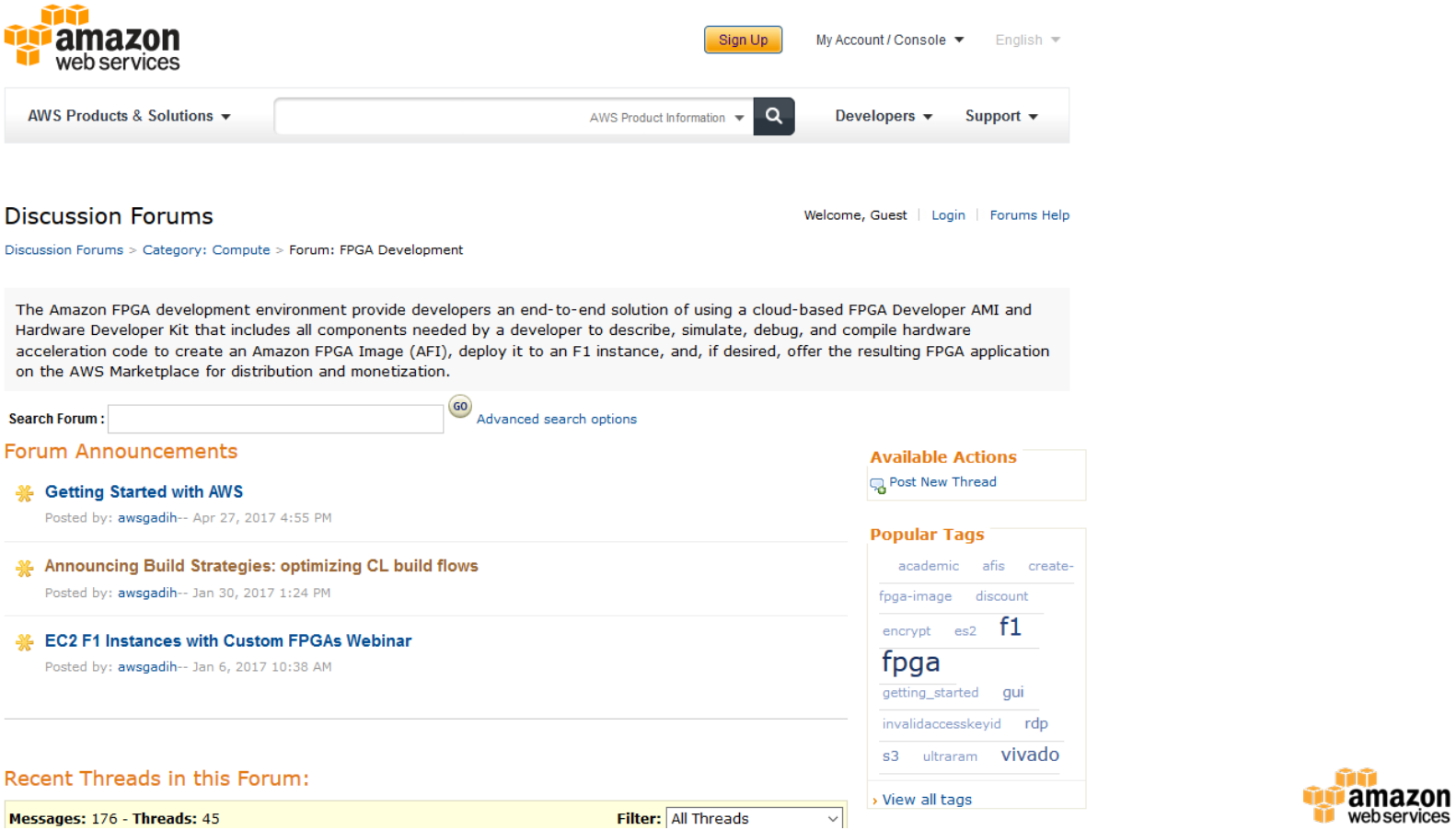
Delivering FPGA Partner Solutions

via AWS Marketplace



AFI is secured, encrypted, dynamically loaded into the FPGA - can't be copied or downloaded

F1 Discussion Forum at forums.aws.amazon.com/



amazon web services

Thank you!

David Pellerin dpelleri@amazon.com